Designing libraries of chimeric proteins using SCHEMA recombination and RASPP

Matthew A. Smith and Frances H. Arnold*

Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, CA, 91125, USA.

* To whom correspondence should be addressed: frances@cheme.caltech.edu

Summary

SCHEMA is a method for designing libraries of novel proteins by recombination of homologous sequences. The goal is to maximize the number of folded proteins, while simultaneously generating significant sequence diversity. Here, we use the RASPP algorithm to identify optimal SCHEMA designs for shuffling contiguous elements of sequence. Our design recombines 5 fungal cellobiohydrolases (CBH1s) to produce a library of more than 390,000 novel CBH1 sequences.

Key words: protein engineering, homologous recombination, SCHEMA, RASPP, chimeragenesis,

Running head: SCHEMA and RASPP

1. Introduction

SCHEMA recombination shuffles sequence elements (blocks) defined by a set of crossover locations in homologous proteins to generate novel chimeric proteins ***(1)*** (*see* **Fig. 1**). Despite that fact that homologous mutations are more conservative than random mutations, a chimera containing many mutations is less likely to be functional than one closer in sequence to one of its parent proteins. SCHEMA recombination seeks to maximize the probability that a library of chimeric proteins will be functional by using structural information to pick crossover locations that minimize disruption of the folded structure. Our metric for disruption is the number of non-native residue-residue contacts, which we refer to as a chimera's SCHEMA energy (E). Minimizing the average SCHEMA energy (<E>) of all the chimeras in a library increases the fraction of functional chimeras ***(2)***. For sequence elements that are contiguous along the polypeptide chain, we developed the RASPP ***(3)*** computational tool to identify crossovers that minimize <E>.

Because chimeric proteins retain sequence elements (e.g. catalytic residues) that are shared among the parents, properly folded chimeras usually retain the overall function of the parents. The new combinations of amino acids in other parts of the protein, however, can lead to significant changes in key properties such as stability ***(4, 5)***, expression level ***(6)***, or substrate specificity ***(7)***. By analyzing a subset of the possible chimera sequences we can build predictive models and identify the chimeras having useful changes in those properties ***(8)***.

In this chapter, we design a SCHEMA library that recombines 5 fungal cellobiohydrolases (CBH1s). We use RASPP to identify optimal libraries having 7 crossover sites (8 blocks). Shuffling these blocks among the 5 homologs generates a recombination library of 5^8 = 390,625 possible sequences. We previously designed a very similar library ***(6)***, and analysis of a subset of chimeras led us to identify chimeric CBH1s that are more stable than any of the 5 parents.

2. Materials

1. A Unix-based computer that can run python scripts (*see* **Note 1**). Python can be downloaded from: http://www.python.org/download/
2. Download and unpack the RASPP toolbox. This is available from: http://cheme.che.caltech.edu/groups/fha/media/schema-tools.zip
3. A multiple sequence alignment of the parental sequences that are to be recombined (*see* **Note 2**). This alignment should be in ALN format (such as that produced by ClustalW), without a header (*see* **Note 3**). As recombination parents, we picked the CBH1 sequences from *C. thermophilum*, *T. aurantiacus*, *H. jecorina*, *A. thermophilum*, and *T. emersonii*, which share approximately 60% sequence identity. These CBH1s have a catalytic domain, a linker and a cellulose-binding

domain. The available crystal structures are for the catalytic domain, thus we only considered this domain for recombination (*see* **Note 4**). To eliminate the possibility of generating unpaired disulfide bonds, we mutated two residues in the *T. emersonii* and *T. aurantiacus* CBH1 sequences to cysteine (*see* **Note 5**). We used ClustalW2 ***(9)*** to align the parental sequences and we named our alignment file 'CBH1-msa.txt'.

4. A PDB structure file of one of the parental sequences (*see* **Note 6**). We used the *T. emersonii* structure, '1Q9H.pdb'.
5. A sequence alignment of one of the parental sequences with the sequence from the PDB structure file (*see* **Note 7**). We used ClustalW2 to align the parental sequences and we named our alignment file 'Temer-1Q9H.txt'.

3. Methods

1. Place the parent sequence alignment file (CBH1-msa.txt), the PDB structure file (1Q9H.pdb) and the PDB alignment file (Temer-1Q9H.txt) in the 'schema-tools' folder.
2. Run the following command (*see* **Note 8**) in the 'schema-tools' directory:

```
python schemacontacts.py -pdb 1Q9H.pdb -msa CBH1-msa.txt -pdbal Temer-
1Q9H.txt -o contacts.txt
```

   This generates a file containing the SCHEMA contacts called 'contacts.txt' (*see* **Note 9**).
3. Run the following command (*see* **Note 10**) in the 'schema-tools' directory:

```
python rasppcurve.py -msa CBH1-msa.txt -con contacts.txt -xo 7 -o opt.txt -
min 15
```

   This RASPP script identifies a set of 8-block candidate libraries with low <E> (*see* **Note 11**). Each block is required to have at least 15 mutations. These libraries are saved to the file 'opt.txt' (*see* **Note 12**) (**Fig. 2**).

4. Pick a library from the results file 'opt.txt' (*see* **Note 13**). In this case, we pick the library with crossover points [33 73 107 175 264 366 415], <E> = 21.2 and <m> = 74.7 (**Fig. 3**).
5. Create a text file called 'CBH1-xo.txt' that contains the crossover points of the chosen library each separated by a space (*see* **Note 14**). The contents of the text file should be the following:
```
33 73 107 175 264 366 415
```
6. Run the following command (*see* **Note 15**) in the 'schema-tools' directory:
```
python schemaenergy.py -msa CBH1-msa.txt -con contacts.txt -xo CBH1-xo.txt -E
-m -o energies.txt
```
This generates a list of all the chimeras in the chosen library along with their SCHEMA energies and number of mutations (*see* **Note 16**). This list is saved to the file 'energies.txt'.
7. At this point we constructed a small chimera test set by substituting each block from each parent into the parental sequence from *T. emersonii*; the corresponding genes were synthesized (*see* **Note 17**). We could also have synthesized the genes encoding a different subset of the library (*see* **Note 18**) or even constructed the entire library (*see* **Note 19**). Before expressing the CBH1 chimeras, we add a linker and cellulose-binding domain to the recombined catalytic domains.

4. Notes

1. The RASPP toolbox 'schema-tools' is written for python 2.6 on a Unix-based system. We recommend using this python release for the RASPP toolbox.
2. As a general rule, when picking sequences for SCHEMA recombination we try to ensure the sequence identity between the homologs is not lower than ~55% if individual genes are to be synthesized. In our experience, recombining sequences with much lower identities results in libraries with a high proportion of non-functional chimeras, even using SCHEMA. (This may

not be a problem if the whole library is constructed and screened for functional chimeras.) The parental sequences are assumed to share the same fold; homologs with >55% identity are likely to have very similar structures. If a structure is available for multiple parental sequences, we confirm they have the same fold by aligning the parental structures.

3. Lines starting with '#' are ignored in the multiple sequence alignment file. Sequence similarity symbols and trailing numbers are also ignored.
4. SCHEMA library designs require a protein structure. If no structural information is available for a parent sequence, but there are structures of homologs, we can use MODELLER to build a structure model ***(10)***. An inaccurate homology model hinders SCHEMA library design; an actual structure is preferred.
5. We assumed but did not verify that broken disulfide bonds are destabilizing. In this case, *C. thermophilum*, *H. jecorina*, and *A. thermophilum* CBH1s have 10 disulfide bonds while *T. aurantiacus* and *T. emersonii* have 9 disulfide bonds. If the cysteines from the missing disulfide bond are in separate sequence blocks, chimeras with unpaired cysteines can result. We avoided this by modifying the parental sequences of *T. aurantiacus* and *T. emersonii* to include the remaining cysteine pair.
6. A structure is necessary to identify the residue-residue contacts. When possible, we pick a high-resolution structure (< 2.0 Å).
7. The sequence of the PDB file can be extracted with the following (run from the 'schema-tools' directory):

   ```
   python -c "import pdb; pdb.get('1Q9H.pdb')"
   ```

   We aligned this PDB sequence with the corresponding parent sequence (*T. emersonii* CBH1) from the parental alignment. The parent sequence must have the same identifier in both alignment files ('Temer') and the identifier of the PDB sequence must be the name of the PDB structure (‘1Q9H’). The PDB sequence can be identical to the parent sequence, but this is not always the case; often the PDB sequence will be truncated or contain several point mutations. In our case we have mutated several of the residues in *T. emersonii* CBH1 to cysteine (*see* **Note 5**).

8. The python script ‘schemacontacts.py’ calculates all of the SCHEMA contacts. Several arguments need to be provided when running this script:
   - ‘-pdb 1Q9H.pdb’: name of the PDB structure
   - ‘-msa CBH1-msa.txt’: name of the parental sequence alignment
   - ‘-pdbal Temer-1Q9H.txt’: name of the PDB sequence alignment
   - ‘-o contacts.txt’: name of an output file to store the contacts
9. Each contact is represented as a pair of residue numbers in ‘contacts.txt’. Numbering is given in terms of both the parental sequence alignment and the PDB sequence alignment.
10. The python script ‘rasppcurve.py’ finds crossover points that minimize the average SCHEMA energy for a library. Several arguments need to be provided when running this script:
    - ‘-msa CBH1-msa.txt’: name of the parental sequence alignment
    - ‘-con contacts.txt’: name of the contacts file
    - ‘-xo 7’: number of crossovers
    - ‘-min 15’: minimum number of non-identical residues in a block (prevents trivial solutions)
    - ‘-o opt.txt’: name of an output file for the results

    This script may take several hours to complete, depending on protein size and computer specifications. Increasing the number of crossovers in a library increases library size and reduces the average number of mutations in a block. The user may want smaller blocks if searching for properties from single point mutations. However, it is harder to find desirable chimeras in larger libraries and increasing the number of blocks increases a library's <E>. We chose to split our 5 parent proteins into 8 blocks.

11. There is a trade-off between the average SCHEMA energy of a library (<E>) and the average number of mutations from the closest parent (<m>), which depends on the relative block sizes (*see* **Fig. 2b**). If all the blocks are evenly sized, <m> is very high but the solution space of possible libraries is very small and so <E> is large. As block sizes become uneven, the solution space of possible libraries increases. This enables RASPP to find libraries with lower <E> but these libraries have lower <m>. RASPP is designed to find low <E> libraries for a range of <m>.
12. Each library is defined by 7 crossover points. The crossover points are given by the first residue of each new fragment (excluding the first fragment, which is always 1) based on the numbering of the parental sequence alignment. The results file 'opt.txt' also gives <E> and the average number of mutations from the closest parent (<m>) for each library.
13. RASPP returns a set of candidate libraries with a range of <m> values. A lower <E> implies more functional chimeras in the library. For moderately sized proteins (250-500 amino acids) we try to pick SCHEMA libraries with <E> less than 30. Protein-specific biochemical and structural knowledge may help users pick from the candidate libraries.
14. Lines starting with '#' are ignored in the crossover file.
15. The python script 'schemaenergy.py' lists the chimeras in a library. Several arguments need to be provided when running this script:
    - '-msa CBH1-msa.txt': name of the parental sequence alignment
    - '-con contacts.txt': name of the contacts file
    - '-xo CBH1-xo.txt': name of the crossover file that defines the library

- '-E -m': specifies that the chimeras should be listed with their E and m values
- '-o energies.txt': name of an output file for the results

16. Chimeras are numbered according to the parental sequence of each block with the numbers ordered from the first block to the last block. Parents are numbered based on the order they appear in the parental sequence alignment. For example, chimera '14221313' has parent 1 as the sequence of its first block, parent 4 as its second block, etc.
17. The fungal CBH1 enzymes have poor heterologous expression in *S. cerevisiae*. Because *T. emersonii* CBH1 expresses much better than the other parents, we analyzed the blocks one at a time in the background of *T. emersonii* CBH1. These chimeras tend to have low SCHEMA energies and they can be easily constructed via overlap extension PCR. Using this 'monomera' approach, we identified stable CBH1 chimeras in a SCHEMA library similar to the one presented here ***(6)***.
18. We pick a subset of the library to analyze. We ensure every block from every parent is represented independently of one another in this subset. This enables us to model the effect blocks have on biochemical properties such as stability ***(5)***.
19. It is possible to construct an entire SCHEMA library in the laboratory by assembling blocks of sequence with specific overhangs ***(11, 12)***. This approach is appropriate for searching for chimeras with specific properties that cannot be predicted from a small library sample.

Acknowledgements

The authors acknowledge funding from the Institute for Collaborative Biotechnologies through grant W911NF-09-D-0001 from the U.S. Army Research Office and The National Central University, Taiwan, through a Cooperative Agreement for Energy Research Collaboration. MAS is supported by a Resnick Sustainability Institute fellowship.

References

1. Voigt, C. A., Martinez, C., Wang, Z.-G., Mayo, S. L., and Arnold, F. H. (2002) Protein building blocks preserved by recombination. *Nat Struct Biol* **9**, 553–558
2. Meyer, M., Hochrein, L., and Arnold, F. H. (2006) Structure-guided SCHEMA recombination of distantly related β-lactamases. *Protein Eng Des Sel* **19**, 563–570
3. Endelman, J., Silberg, J., Wang, Z., and Arnold, F. H. (2004) Site-directed protein recombination as a shortest-path problem. *Protein Eng Des Sel* **17**, 589–594
4. Romero, P., Stone, E., Lamb, C., Chantranupong, L., Krause, A., Miklos, A., Hughes, R., Fechtel, B., Ellington, A. D., Arnold, F. H., and Georgiou, G. (2012) SCHEMA-designed variants of human arginase I and II reveal sequence elements important to stability and catalysis. *ACS Synth Biol* **1**, 221–228
5. Li, Y., Drummond, D. A., Sawayama, A. M., Snow, C. D., Bloom, J. D., and Arnold, F. H. (2007) A diverse family of thermostable cytochrome P450s created by recombination of stabilizing fragments. *Nat Biotechnol* **25**, 1051–1056
6. Heinzelman, P., Komor, R., Kanaan, A., Romero, P. A., Yu, X., Mohler, S., Snow, C., and Arnold, F. H. (2010) Efficient screening of fungal cellobiohydrolase class I enzymes for thermostabilizing sequence blocks by SCHEMA structure-guided recombination. *Protein Eng Des Sel* **23**, 871–880
7. Otey, C. R., Landwehr, M., Endelman, J. B., Hiraga, K., Bloom, J. D., and Arnold, F. H. (2006) Structure-guided recombination creates an artificial family of cytochromes P450. *PLoS Biol* **4**, e112
8. Heinzelman, P., Romero, P. A., and Arnold, F. H. (2013) Efficient sampling of SCHEMA Chimera Families for Identification of Useful Sequence Elements. In: Keasling, A (ed) Methods in Enzymology: Methods in Protein Design, Elsevier Ltd, Oxford, U.K.
9. Larkin, M. A., Blackshields, G., Brown, N. P., Chenna, R., McGettigan, P. A., McWilliam, H., Valentin, F., Wallace, I. M., Wilm, A., Lopez, R., Thompson, J. D., Gibson, T. J., and Higgins, D. G. (2007) Clustal W and Clustal X version 2.0. *Bioinformatics* **23**, 2947–2948
10. Eswar, N., Webb, B., Marti-Renom, M. A., Madhusudhan, M. S., Eramian, D., Shen, M.-Y., Pieper, U., and Sali, A. (2007) Comparative Protein Structure Modeling Using MODELLER. *Curr Protoc Protein Sci* **50**, 2.9.1–2.9.31
11. Hiraga, K., and Arnold, F. (2003) General method for sequence-independent site-directed chimeragenesis. *J Mol Biol* **330**, 287–296
12. Farrow, M. F., and Arnold, F. H. (2010) Combinatorial Recombination of Gene Fragments to Construct a Library of Chimeras. *Curr Protoc Protein Sci* **62**, 26.2.1–26.2.20

Figure captions

Fig. 1. SCHEMA recombination. Homologous protein sequences are split into blocks at fixed crossover locations. These blocks are shuffled to generate novel chimeric proteins.

Fig. 2. Libraries returned by RASPP. (**a**) The contents of 'opt.txt', which lists the crossover locations of candidate libraries identified by RASPP. (**b**) A graph of the possible libraries plotting average SCHEMA energy (<E>) of each library against the average number of mutations (<m>). The trade-off between <E> and <m> is apparent. The chosen library is highlighted with an arrow.

Fig. 3. Visualizing the chosen RASPP design. (**a**) The multiple sequence alignment of the parent CBH1s with each of the 8 blocks highlighted in a different color. (**b**) The blocks highlighted on the CBH1 structure '1Q9H.pdb'.